การประชุมวิชาการเครือข่ายวิศวกรรมเครื่องกลแห่งประเทศไทยครั้งที่ 18
18-20 ตุลาคม 2547 จังหวัด ขอนแก่น

# การตรวจสอบและหาสาเหตุข้อผิดพลาดในเครื่องจักรด้วยโปรเจคชั่นเน็ทเวิร์ค

# Projection Network for Machine Fault Detection and Diagnosis

ชัยยากร จันทร์สุวรรณ์[1] C. James Li[2]
สถาบันค้นคว้าและพัฒนาเทคโนโลยีการผลิตทางอุตสาหกรรม คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเกษตรศาสตร์ กรุงเทพมหานคร 10900
Department of Mechanical, Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute
Troy, New York 12180, USA[2]
โทร 0-29428567-70[1] โทรสาร 0-29428571[1] E-mail: fengckj@ku.ac.th[1], lic3@rpi.edu[2]

Chaiyakorn Jansuwan[1]* C. James Li[2]
Research and Development Institute of Industrial Production Technology, Faculty of Engineering,
Kasetsart University, Bangkok 10900[1]
Department of Mechanical, Aerospace and Nuclear Engineering, Rensselaer Polytechnic Institute
Troy, New York 12180, USA[2]
Tel: 0-29428567-70[1] Fax: 0-29428571[1] E-mail: fengckj@ku.ac.th[1], lic3@rpi.edu[2]

**Abstract**

Machine fault is a condition when a machine is inhibited from proper operation due to defect in its components, which can lead to a more serious problem such as machine breakdown. Therefore, it is imperative to correctly detect and identify fault in its early stage to avoid costly consequences, especially in the critical machines. This paper introduces a projection network and its utility for machine fault detection and diagnosis. Projection network is essentially a dynamic network sandwiched by two linear mappings. The dynamic network forms a memory space for groups of faults and is connected to the real space through input and output mappings. After properly trained, projection network is able to correctly detect and diagnose faults from unseen data. An example of projection network utility is also illustrated for the case of high pressure air compressor fault detection.

## 1. Introduction

Today industrial machinery has become more capable and more automated than ever. These sophisticated machines work very efficiently at an incredible speed. They, however, also require a better maintenance plan in order to prevent costly machine breakdown. Such plan must be able to not only completely eliminate breakdown but also fully utilize components' useful life. This means that a kind of device capable of detecting incipient fault in machine components must be employed. This concept is the heart of a Condition-Based Maintenance (CBM) programs – also known as predictive maintenance program.

Figure 1 shows a diagram of a typical CBM system. First, operating data from a data acquisition system is preprocessed and then used to assess the health of the system. This includes the detection and identification (diagnosis) of any fault that may occur in the monitored system. If any kind of anomaly is detected the information will be passed to a prognostic module for an estimation of the health status at some future time for the involved parts. A maintenance plan is then arranged accordingly.

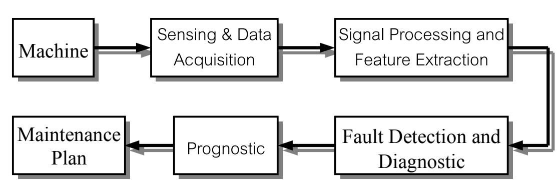


Figure 1. Typical Condition-Based Maintenance System

A crucial step in a CBM system is fault detection and diagnosis since it is the ground for subsequent procedures. Failing to correctly detect and identify incipient faults will lead to

* Corresponding author

erroneous maintenance plans, which may, as a result, cause substantial losses.

A problem of detecting and identifying fault from machine operating data may be viewed as a pattern classification problem, which can, therefore, be solved with existing pattern classifiers, which are mostly static in nature – for example a Bayes classifier, a decision tree classifier and a neural network classifier.

This paper, however, give an alternative view of the pattern classification problem as a construction of a potential surface in feature space that optimally fits the sample data profile. This way the potential surface can be pictured as a surface with the number of bowl-shaped depressions at least equal the number of classes. Further imagine that if one puts a ball (a sample data) on the potential surface, the ball will roll down the slope and finally rest at the bottom of one of the basins (a class representative). That is, a sample data will be assigned its class after the process. Therefore, this process is dynamic in nature.

This study employs a recurrent (dynamic) type neural network because it can approximate a wide class of potential surfaces, and a parameter training process to find the appropriate parameters for the network. An example of the dynamic neural network is a fully connected recurrent neural network. The fully connected recurrent neural network, however, is very difficult to work with due to the lack of an efficient training algorithm, guideline of its utility and knowledge about the network's properties.

In this study, a form of a high-order dynamic neural network called projection network [1] is selected. Projection network is a simplified, alternative implementation of a projection algorithm, which is derived from the normal form equations for a multiple Hopf bifurcation. This form of dynamic network is less complex than the fully connected recurrent neural network and therefore is easier to work with in terms of understanding the behavior of the system and the training process (note that training is a major stumbling block for the utilization of the fully connected recurrent neural network) yet has rich enough dynamics necessary for the application at hand. The easier training process for the projection network is derived from the fact that the network has a less complex structure, which also means a better-behaved potential surface.

A projection network has an architecture similar to that of an associative memory network and this gives it one nice property, the ability to efficiently deal with incomplete/corrupted data since the input pattern will go through the dynamic evolution process (filtering process) until it reaches the final state (filtered/clean output pattern). This is particularly useful for the real-world applications where incomplete/corrupted data is common. This also implies a non-ambiguous output of the projection network, which makes it suitable for classification task.

Another advantage of the projection network is its well-understood qualitative properties such as equilibrium points and their stability conditions. An understanding of these properties will be helpful for the designing of the projection network so that it carries out the intended tasks robustly.

In this study, the projection network is first studied for the relationship between its parameters and its properties including stability, attractor location, basin shape, etc. Then, guidelines and algorithms for structure and parameter learning are established for classification. Subsequently, a case study of a high pressure air compressor fault diagnosis is given.

**2. Projection Network**

A projection network is a 3rd order dynamic neural network and a simplified, alternative form of a projection algorithm, which was originally proposed by Baird and Eeckman [1] as an attempt to model an oscillatory associative memory – the type that could occur in biological cognitive operations. Figure 2 shows a block diagram of the projection network. It shows that the projection network architecture is a dynamic memory network sandwiched by two linear mappings.

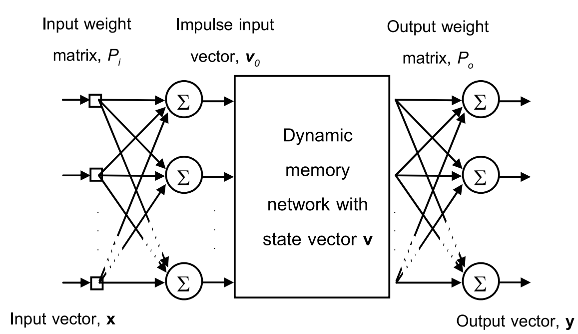


Figure 2 Projection Network

The feed-forward part includes two linear projections: one from input coordinates to memory coordinates and another from the memory coordinates to output coordinates. The embedded nonlinear dynamic memory network is governed by a set of differential equations

$$\dot{v}_i = \alpha_i v_i - v_i \sum_{j=1}^{n} a_{ij} v_j^2 , \quad (1)$$

where $v$'s are the state variables, $A$ = [$a_{ij}$] is a weight matrix and *Alpha* = [$\alpha_i$] is a linear gain vector.

A projection network operates as follows. First an input vector is projected as an initial condition into the memory coordinate, i.e. $\mathbf{v}_0 = P_i \mathbf{x}$, where the dynamic evolution (relaxation) takes place. This means the memory network's state evolves over time. If the network is stable, the state will ultimately rest at the local minimum point of the basin where $\mathbf{v}_0$ started. The final state is then projected out of the memory coordinates and into the output space, $\mathbf{y} = P_o \mathbf{v}$.

A local minimum, also known as an attractor, is a stable equilibrium point – a point in state space where all the derivatives vanish. In this study, a class of axis equilibrium points is of special interest due to its programmable location and stability. An axis equilibrium point is an equilibrium point where all but one of the components takes on a zero value. Axis equilibrium point $\tilde{v}_s$ on axis $s$ is obtained by setting the other components $v_j$, $j \neq s$, to zeros, which leaves

$$0 = \tilde{v}_s \left( \alpha_s - a_{ss} \tilde{v}_s^2 \right)$$

or

$$\tilde{v}_s = \sqrt{\frac{\alpha_s}{a_{ss}}} \tag{2}$$

Next, stability condition for this axis attractor can be obtained by following a standard method in dynamic system analysis. This starts with determining the Jacobian matrix $U$ of the right-hand-side of equation (1) and then evaluating the eigenvalues of the Jacobian matrix at the equilibrium point of interest. If all eigenvalues are negative, then the fixed point is stable. The elements of the Jacobian matrix are

$$U_{ii} = \alpha_i - 3a_{ii} v_i^2 - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} v_j^2 \tag{3a}$$

and

$$U_{ij} = -2a_{ij} v_i v_j \tag{3b}$$

Considering an axis fixed point $\tilde{v}_s = \sqrt{\frac{\alpha_s}{a_{ss}}}$ on axis $s$, Equation (3b) becomes

$$U_{ij} = 0 \tag{4a}$$

since either $v_i$ or $v_j$ is zero. There are two possible cases for Equation (3a):

case i) $i \neq s$, (i.e., $v_i = 0$, $v_j = 0$ *except* $v_s$)

$$\begin{aligned} U_{ii,i \neq s} &= \alpha_i - a_{is} \tilde{v}_s^2 \\ &= \alpha_i - a_{is} \frac{\alpha_s}{a_{ss}} \end{aligned} \tag{4b}$$

case ii) $i = s$,

$$\begin{aligned} U_{ss} &= \alpha_s - 3a_{ss} \tilde{v}_s^2 \\ &= \alpha_s - 3a_{ss} \frac{\alpha_s}{a_{ss}} = -2\alpha_s \end{aligned} \tag{4c}$$

Since the off-diagonal elements of the Jacobian matrix are all zeros (Equation 4a), the diagonal elements are then the eigenvalues. For a stable axis fixed point $\tilde{v}_s$, Equations (4b-c) have to take on negative values. Therefore, the axis fixed point $\tilde{v}_s$ is stable if and only if,

from (4c), $$\alpha_s > 0 \tag{5a}$$

and, from (4b), $$\alpha_i - a_{is} \frac{\alpha_s}{a_{ss}} < 0$$

or, because of (5a), $$\frac{\alpha_i}{\alpha_s} < \frac{a_{is}}{a_{ss}} \tag{5b}$$

Equation (5a) requires all elements of the linear gain vector *Alpha* to be positive, which consequently forces the diagonal elements, *a's*, of the weight matrix *A* to take on positive values to ensure the real axis fixed points according to Equation (2). Moreover, these conditions in conjunction with Equation (5b) also make the off-diagonal elements of the weight matrix *A* to take on positive values. As a result, the stability conditions are represented by Equation (6) below.

$$\frac{\alpha_i}{\alpha_s} < \frac{a_{is}}{a_{ss}} \tag{6a}$$

and

$$\alpha_i, a_{ij} > 0, \ \forall \, i, j \tag{6b}$$

By enforcing these stability conditions, the axis equilibrium points will become axis attractors, each of which has its associated basin of attraction.

### 3. Machine Fault Detection and Diagnosis

The key idea of utilizing a projection network for classification is that only patterns of the same class are placed into the same basin of attraction in memory coordinates. This is done via the input weight matrix, $P_i$. A dynamic memory network then will recover each pattern's associated stored memory or the pattern's associated class. In this case, the output mapping can be used to project a recovered pattern into the output space for easier interpretation. This, however, is not necessary for classification purposes.

To implement a classifier based on projection network, one first sets up the dynamic memory network using the previously derived conditions so that the memory network has its number of attractors equal to the number of classes of the data at hand. Then the input mapping is used to project the patterns from input space onto memory coordinates as close as possible to their associated attractors. This can be done using a single shot calculation of a weighted pseudo-inverse operation. By setting the target matrix to the attractor locations, **V**, the input weight matrix $P_i$ can be determined using the weighted pseudo-inverse as follows.

$$P_i \mathbf{XW} = \mathbf{VW} \tag{7}$$
$$P_i \mathbf{XWX}^{\mathbf{t}} = \mathbf{VWX}^{\mathbf{t}}$$

The required linear transformation is

$$P_i = \mathbf{VWX}^{\mathbf{t}} (\mathbf{XWX}^{\mathbf{t}})^{-1} \tag{8}$$

where **X** is an input matrix of dimension $N_i \times N$, **V** is of dimension $N_c \times N$, $P_i$ is of dimension $N_c \times N_i$ and **W** is a diagonal, weight matrix of dimension $N \times N$. $N_i$, $N_c$, and $N$, are the number of inputs, the number of classes and the number of total patterns in the dataset respectively. An equal-class weighted pseudo-inverse is used to eliminate the effect of class size discrepancies. To realize this, one sets the weights of patterns of different classes inversely proportionally to their class sizes.

**4. Training**

In training the projection network for classification, there are two main modules: the input mapping module and the dynamic memory network module. Since the constraints are different and each module has a number of parameters, we opted for a sequential training strategy, i.e. train one module first, then switch to another module and keep doing so until the result satisfies a stopping criterion, which, in this case, is a minimum square error (MSE) function. Since the output of the projection network classifier is one of the class representatives, the correctly classified samples will all have the same outputs and different from the wrongly classified samples. That is only the wrongly classified samples contribute to the MSE objective function. Therefore, a combined use of MSE objective function and dynamic memory network, in a way, transforms the conventional MSE function to a superior minimum-classification-error (MCE) like objective function.

The training procedures for the two modules are also different. In input mapping module training, there is no constraint about the values the input weights might take. As such, an unconstrained optimization algorithm can be used. In this study, a BFGS quasi-Newton method was employed. The training of the dynamic memory network module is quite different from that of the input mapping module due to the stability constraints, i.e., Equation (6). This leads to the need of a constrained optimization algorithm. This study employed a sequential quadratic programming (SQP) algorithm for this task.

**5. HPAC fault diagnosis**

A High-Pressure Air Compressor (HPAC) is selected to demonstrate the use of projection network classifier for machine fault diagnosis. HPACs are found in many industrial and military applications, such as in the automotive industry, blow molding, metal forming, diesel and gas turbine starting systems, offshore drilling, power generation, and maritime and Navy applications.

The HPAC in question is a reciprocating, four-stage, vertical, single action unit whose nominal operating pressure is 4,000 psi. The compressor is oil-lubricated and water-cooled and driven by a 50 HP, 900 RPM, three-phase, 440 Volt squirrel cage induction motor. The first and third stages are on one crank throw and the second and fourth stages are on the opposite throw.

The frequently encountered problems with the HPAC are the internal leakages, namely the suction and discharge valve leakages. These faults degrade the performance of the compressor. This case study focuses on the diagnosis of faults in the third stage, namely suction and discharge valve leakages and cylinder pitting. The first two were identified as major failure modes in a review of historical data [3]. Consequently, experimental data was first collected under the baseline condition and then by deliberately introducing a leaky valve disk into either the suction or discharging port on the third stage. Cylinder pitting was an unexpected encounter in the course of the experiments.

Before using the dataset to design the classifier, the dataset was screened and some invalid data points such as ones that contain zero or negative values were removed. Subsequently the outlier elimination algorithm was employed to further remove outliers. From this conditioned dataset, five 80/20 training and testing sets were randomly generated and used to design and evaluate the projection network supervised classifier. The results were recorded in terms of correct classification rate and shown in Table 1. The average correct classification rates for the training and testing sets are 99.9 % and 99.76 % respectively. Table 2 shows the classification results for the HPAC in terms of confusion matrix of the testing set. It should be noted that there is only 0.82 % “false alarm” for the cylinder pitting and corrosion fault and that there is no missed-detection.

Table 1. Classification Result for HPAC

| Dataset Number | Correct Classification Rate (%) | |
|---|---|---|
| | Training set | Testing set |
| 1 | 99.85 | 100 |
| 2 | 99.93 | 99.7 |
| 3 | 99.85 | 99.7 |
| 4 | 99.93 | 99.7 |
| 5 | 99.93 | 99.7 |
| Average | 99.90 | 99.76 |

Table 2. Confusion Matrix

| Actual Fault | Assigned Fault | | | |
|---|---|---|---|---|
| | Baseline | $3^{rd}$ stage DVF | $3^{rd}$ stage SVF | Pitting |
| Baseline | 99.18 | | | 0.82 |
| $3^{rd}$ stage DVF | | 100 | | |
| $3^{rd}$ stage SVF | | | 100 | |
| Cylinder pitting | | | | 100 |

## 6. Conclusion

This paper describes the projection network, its properties, and its utility for pattern classification. This study performed analysis of the network’s equilibrium points, stability conditions and established its parameter initialization and optimization methods. It also showed that a projection network actually behaves like a minimum-classification-error (MCE) classifier in spite of its least square objective function. In addition, a projection network classifier was established to diagnose faults in a high-pressure air compressor with satisfactory results. It was concluded that the projective network is a viable alternative to existing classifiers.

## References

[1] Baird, B. and F. Eeckman, “A Normal Form Projection Algorithm for Associative Memory,” Associative Neural Memories, M. H. Hassoun, Editor, Oxford university press, New York, pp. 135 – 166, 1993.

[2] Chaiyakorn Jansuwan, “Dynamic Projection Network for Associative Memory and Pattern Classification and its Utility for Mechanical Diagnosis,” Doctoral Thesis, Rensselaer Polytechnic Institute, Troy, New York, 2003.

[3] NAVSEA, “Systems and Equipment Maintenance Monitoring for Surface Ships (SEMMSS) FF1052 High-Pressure Compressed Air System Study," Final Report, NAVSEA N00024-84-D-4312, Technical Instruction 050-501, June 30, 1986.